



Desarrollo y gestión ágil de proyectos Python

Lección 3: Artefactos de Scrum

ÍNDICE

Artefactos de Scrum	3
Presentación y objetivos	3
1. Artefactos de Scrum	4
2. Pila del producto	5
3. Pila del sprint	8
4. Incremento de sprint	10
5. Burn-Down	12
6. Historias de usuario.....	16
6.1. Información en las historias de usuario.....	18
7. Puntos clave.....	21

Artefactos de Scrum

«La agilidad es una actitud no una técnica» ~ Alistair Cockburn

PRESENTACIÓN Y OBJETIVOS

Para aprender a trabajar en un entorno ágil con Scrum, debes conocer todos los artefactos que forman parte del entorno de trabajar, la indiosincrasia de cada uno de ellos y cómo aplicarlos para obtener los mejores resultados posible.



Objetivos

En esta lección aprenderás:

- Cuáles son los principales artefactos de Scrum y cómo trabajar con ellos.
- *Cómo las historias de usuario representan los requisitos del software en un marco de trabajo ágil como Scrum*

1. ARTEFACTOS DE SCRUM

Los artefactos de scrum son sus herramientas, sus bloques de construcción elementales.

Podemos destacar tres artefactos clave para el funcionamiento del marco estándar:

- | **Pila del producto / product backlog:** Registra y prioriza los requisitos desde el punto de vista del cliente. Empieza con una visión inicial del producto y crece y evoluciona durante el desarrollo. Los requisitos suelen denominarse **historias de usuario**, que se descomponen en **tareas** de menor tamaño.
- | **Pila del sprint / sprint backlog:** Refleja los requisitos desde el punto de vista del equipo. Es una lista de los trabajos a realizar durante un sprint para generar el **incremento** previsto.
- | **Incremento:** resultado de cada sprint.

Otros artefactos que también son habituales:

- | Gráfico de avance o **burn down** chart: indica el trabajo pendiente y la velocidad a la que se están completando las tareas para deducir si se completarán todas en el tiempo estimado. El equipo lo actualiza a diario.
- | Gráfico de producto o **burn up** chart: si el gráfico de avance mide lo que falta, el de producto mide cuánto se ha construido o completado.

En la práctica se suele usar el gráfico burn down en mayor medida.

Un elemento esencial de Scrum, pero que a día de hoy no es considerado un artefacto como tal es la **historia de usuario / user story**.

2. PILA DEL PRODUCTO

La pila del product (product backlog) es una lista que describe todos los requisitos del proyecto.

Puesto que la pila del producto representa los requisitos del cliente podemos decir que es el inventario de funcionalidades, mejoras, tecnología y corrección de errores que deben incorporarse al producto a través de los sucesivos sprints.

Representa todo aquello que esperan cliente, usuarios y demás partes interesadas.

Lo más común es referirse a las entradas de esta pila como historias de usuario.

Algunos ejemplos de historias de usuario presentes en la pila de product pueden ser:

- | Ofrecer a los usuarios la consulta de archivos publicados por un determinado miembro de la plataforma.
- | Consultar los pedidos realizados por un vendedor en un rango de fechas.
- | Ofrecer la consulta de un archivo a través de un API web.

La característica esencial de este artefacto es que contiene información viva, en continua evolución, y que más que un documento de requisitos es una herramienta que facilita la comunicación de información al equipo.

Al comenzar el proyecto la lista contiene unos pocos requisitos, aquellos conocidos y mejor entendidos en ese momento, porque se ampliará y modificará conforme avanza el desarrollo.

Este carácter dinámico permite que el producto se adapte a circunstancias cambiantes.

Una vez que la pila del producto tiene historias suficientes para realizar un primer sprint es suficiente para empezar.

A partir de entonces el propietario del producto mantendrá las historias de la pila ordenadas según su prioridad. El nivel de urgencia vendrá marcado por lo necesaria y valiosa que sea cada funcionalidad.

Por otro lado, el grado de concreción de las historias de usuario deberá ser proporcional a su prioridad. Las más prioritarias deben estar lo bastante detalladas como para descomponerse en tareas y pasar al siguiente sprint.

Las tareas de priorización, detalle y preestimación de las historias, previas al sprint, se suelen llamar *refinado* (este nombre viene dado porque han pasado por el evento de refinamiento que veremos más adelante) o *preparación*.

El propietario del producto y los desarrolladores pueden realizarlas en cualquier momento, de forma colaborativa, pero nunca deberían consumir más del 10% de la capacidad de trabajo del equipo.

Más tarde los desarrolladores realizarán una segunda estimación más detallada, en la *reunión de planificación del sprint* (que explicaremos más adelante).

La responsabilidad de estimar el esfuerzo previsible para cada elemento de la posterior lista de tareas es de los desarrolladores que harán el trabajo, como veremos más adelante.



Presta atención

Las historias de usuario de la pila del producto que pueden ser incorporadas a un sprint se denominan preparadas o ready.

Se aplica este término (o similar) para indicar que el propietario del producto y el equipo están de acuerdo en que la historia está lista para ser seleccionada para el sprint.

Es decir: está definida, preestimada, es asumible por sí misma en un único sprint, y se han establecido los criterios para considerarla terminada, así como la persona responsable de verificar que se cumplen.

Para confeccionar y mantener la pila de producto, lo ideal es emplear medios simples, conocidos y compartidos por todo el equipo. Es un radiador de información útil y una herramienta que facilita la comunicación directa. Se pueden anotar las historias de usuario, por ejemplo, en etiquetas adhesivas sobre tableros, ordenándolas según su prioridad; o emplear una herramienta de gestión con la que todo el equipo esté familiarizado, tipo Jira.

Así pues, el objetivo de la pila del producto es describir el estado que tendrá el producto en el futuro y que dibuja la visión compartida por el equipo para planificar.

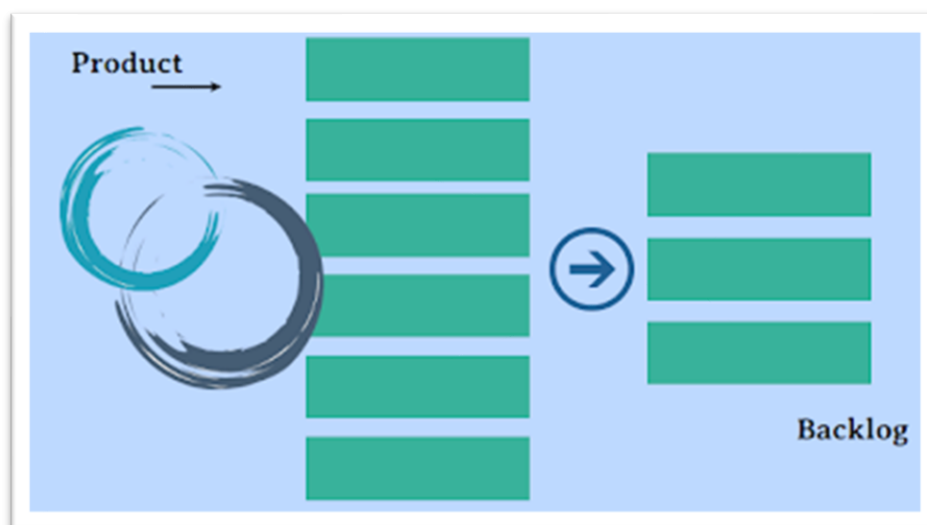


Ilustración 2.1. Pila del producto

3. PILA DEL SPRINT

La pila del sprint (sprint backlog) es la lista específica de elementos cogidos del product backlog que se deben completar en un Sprint.

El equipo debe comprometerse a finalizar en un Sprint todo el listado resultante de la planificación del Sprint (Sprint Backlog), llevando a cabo la implementación de cada una de las User Stories seleccionadas.

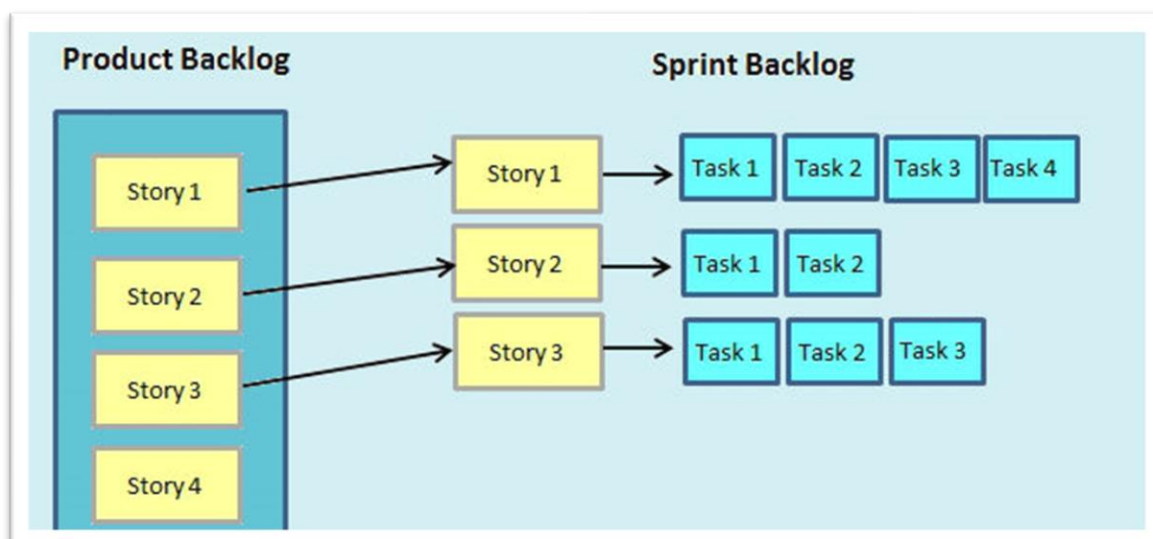


Ilustración 3.1. Pila del Sprint

La pila del sprint o sprint backlog es la lista de todas las tareas necesarias para construir las historias de usuario que se van a realizar en un sprint. En ella las historias de usuario se descomponen en unidades de tamaño adecuado para monitorizar el avance a diario, así como para identificar riesgos y problemas sin procesos de gestión complejos.

Todo el equipo colabora en la confección de esta pila, durante la reunión de planificación del sprint, indicando para cada tarea el esfuerzo previsto para realizarla. Para calcular el esfuerzo de cada ítem en puntos o tiempo es habitual emplear técnicas como la estimación de póquer (que explicaremos más adelante). Los items de mayor tamaño se dividen en otros, de modo un solo nunca dure más de un día de trabajo.

Si la pila del producto es territorio del propietario del producto, la pila del sprint es territorio del equipo. Sus miembros son los únicos que pueden modificarla durante el sprint.

Proporciona además comunicación visual directa y sobre ella el equipo revisa a diario el avance del sprint. Lo ideal es que se encuentre en un tablero o pared en el mismo espacio físico donde se trabaja, para que sea visible por todos. Aunque con la tendencia del teletrabajo y la existencia de equipos geográficamente distribuidos cada vez es más habitual el uso de herramientas como Jira.

SPRINT																
INICIO	DURACIÓN		J	V	L	M	X	J	V	L	M	J	V	L		
1	1-mar-07	12														
			1-mar	2-mar	5-mar	6-mar	7-mar	8-mar	9-mar	12-mar	13-mar	15-mar	16-mar	19-mar		
			23	23	19	16	16	13	9	9	9	9	9	9		
			276	246	216	190	178	158	110	110	110	110	110	110		
SPRINT BACKLOG			ESFUERZO													
Tarea	Estado	Responsal														
Descripción de la tarea 1	Terminada	Luis	16	16	16	16	16	16								
Descripción de la tarea 2	Terminada	Luis	12	8												
Descripción de la tarea 3	Terminada	Luis	4	4	4	4	4									
Descripción de la tarea 4	Terminada	Elena	8	4												
Descripción de la tarea 5	Terminada	Elena	16	16	4											
Descripción de la tarea 6	Terminada	Elena	6	6	2											
Descripción de la tarea 7	Terminada	Antonio	16	4												
Descripción de la tarea 8	Terminada	Antonio	16	16	20	12	4									
Descripción de la tarea 9	Terminada	Antonio	12	2												
Descripción de la tarea 10	En curso	Luis	12	12	12	12	12	12	12	12	12	12	12	12		
Descripción de la tarea 11	Pendiente	Luis	8	8	8	8	8	8	8	8	8	8	8	8		
Descripción de la tarea 12	Terminada	Luis	14	14	14	14	14	14								

Ilustración 2.2. Ejemplo de pila de sprint en excel

Lo apropiado es utilizar el formato más cómodo para todos, teniendo en cuenta los siguientes criterios:

- Debe incluir sólo la información necesaria:
- Lista de tareas.
- Persona responsable de cada tarea.
- Estado en el que se encuentra y «esfuerzo» que queda para completarla.
- Debe servir de medio para registrar, en cada reunión diaria del sprint, el esfuerzo que le queda a cada tarea.
- Debe facilitar la consulta y la comunicación diaria y directa del equipo.

La pila del sprint debe definir y estar alineada con el objetivo del sprint, que marca un hito en el avance hacia la visión del producto.

4. INCREMENTO DE SPRINT

A cada una de las iteraciones de un proyecto Scrum se le denomina Sprint. El Sprint no es un artefacto, sino un evento; de hecho es el evento clave de Scrum.

Un sprint es cada uno de los ciclos o iteraciones que vamos a tener dentro de dentro de un proyecto Scrum. Los Sprints permiten tener un ritmo de trabajo con un tiempo prefijado, siendo la duración habitual de un Sprint entre dos semanas y un máximo de dos meses. En cada Sprint o cada ciclo de trabajo se consigue un entregable o incremento del producto, que aporta valor al cliente.

Incremento de Sprint es la suma de todas las tareas que se han completado desde la última versión de producto entregada. Es responsabilidad del equipo asegurarse de que todo lo que se incluye en un incremento funciona y está listo para ser entregado o “ponerse en producción” aunque depende del product owner decidir cuándo se hace.

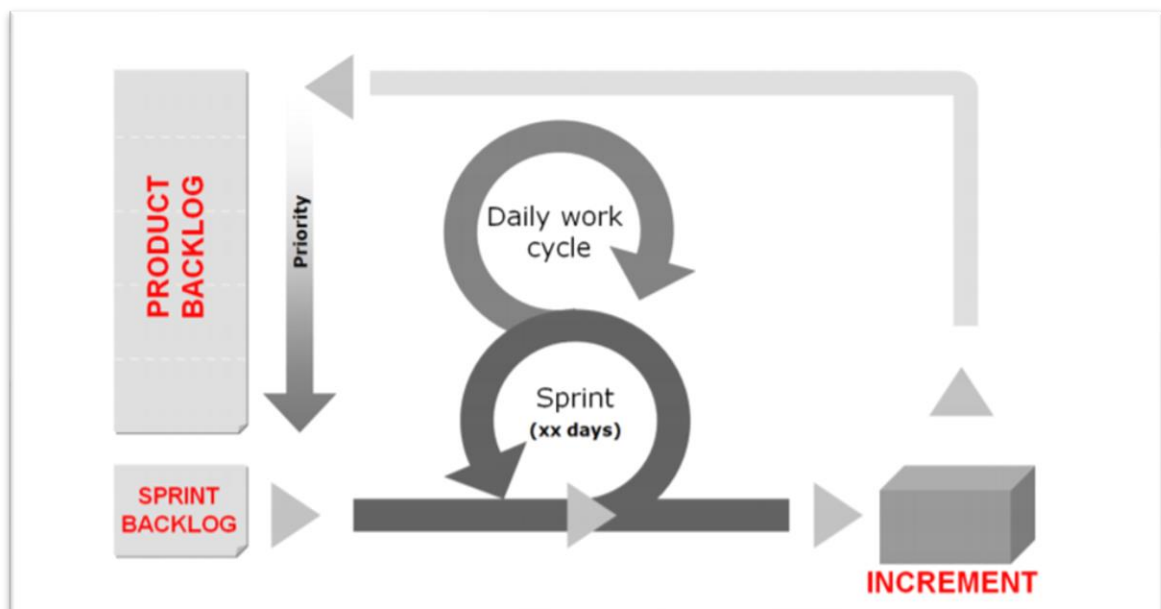


Ilustración 3.1. Sprint e incremento de Sprint

El «incremento» es la parte de producto producida en un sprint y que se encuentra en condiciones de ser entregada al cliente; es decir: terminada, probada y operativa. No se deben considerar incrementos a prototipos, módulos ni a partes pendientes de pruebas.

Idealmente, en scrum:

- | Cada elemento de la pila del producto se refiere a funcionalidades entregables, no a trabajos internos del tipo «diseño de la base de datos».
- | Se produce un incremento en cada iteración / sprint.

Sin embargo, el primer sprint suele ser una excepción. Se suele denominar sprint cero cuando tiene objetivos como contrastar la plataforma y el diseño, que son necesarios al comenzar algunos proyectos. Implican trabajos de diseño o desarrollo de prototipos, para contrastar las herramientas o métodos de trabajo previstos.



Importante

Idealmente, en scrum, Se produce un incremento en cada iteración / sprint.

Si la parte desarrollada requiere documentación, o procesos de validación y verificación documentados, éstos también tienen que estar realizados para considerar al incremento terminado o done, es decir: entregable al cliente.

El objetivo del incremento es cumplir con las medidas de calidad que requiere el producto. La "definición de hecho", debe ser conocida y compartida por todo el equipo, y el incremento no se considera terminado hasta que no se alcanza el criterio de "hecho".

5. BURN-DOWN

Un gráfico **burn-down** o diagrama de quemado es una representación gráfica del trabajo por hacer en un sprint.

Normalmente el trabajo remanente (o sprint backlog) se muestra en el eje vertical en puntos de historia (story points). En el ejemplo de la imagen, para este sprint el equipo se ha comprometido a entregar 85 puntos.

El tiempo en el eje horizontal. En el ejemplo de la imagen, cada sprint tiene 15 días de duración.

Así, el diagrama representa una serie temporal del trabajo pendiente.

Este diagrama es útil para predecir cuándo se completará todo el trabajo.

En la imagen la línea roja representa como deberían ir terminando los trabajos de manera ideal, mientras la línea azul representa el trabajo real mediante una representación del trabajo remanente.

Aunque, este tipo de diagrama ha ido perdiendo importancia dentro de La Guía Oficial de Scrum desde las versiones de 2013 y 2016, muchas organizaciones lo siguen usando ya que, junto al tablero o sprint board es una representación muy visual del estado actual de los trabajos.

Esta herramienta visual es muy útil para predecir cuándo se completará todo el trabajo.

En la Daily Scrum, el equipo de desarrollo actualiza el Sprint Burn Down y traza el trabajo restante del día.

Podemos decir que es casi una herramienta "**imprescindible**" para un equipo Scrum por las siguientes razones principales:

1. Monitoreo del alcance del proyecto
2. Mantener al equipo funcionando a tiempo
3. Comparar el trabajo planificado con la progresión del equipo.

Para crear este gráfico hay que tener en cuenta las siguientes consideraciones:

- | Es necesario determinar cuánto trabajo queda sumando las estimaciones de Sprint Backlog todos los días del Sprint.
- | La cantidad de trabajo restante para un Sprint es la suma del trabajo restante para todo el Backlog de Sprint.
- | Es imprescindible mantener un registro de estas sumas por día y usarlas para crear un gráfico que muestre el trabajo que queda con el tiempo.

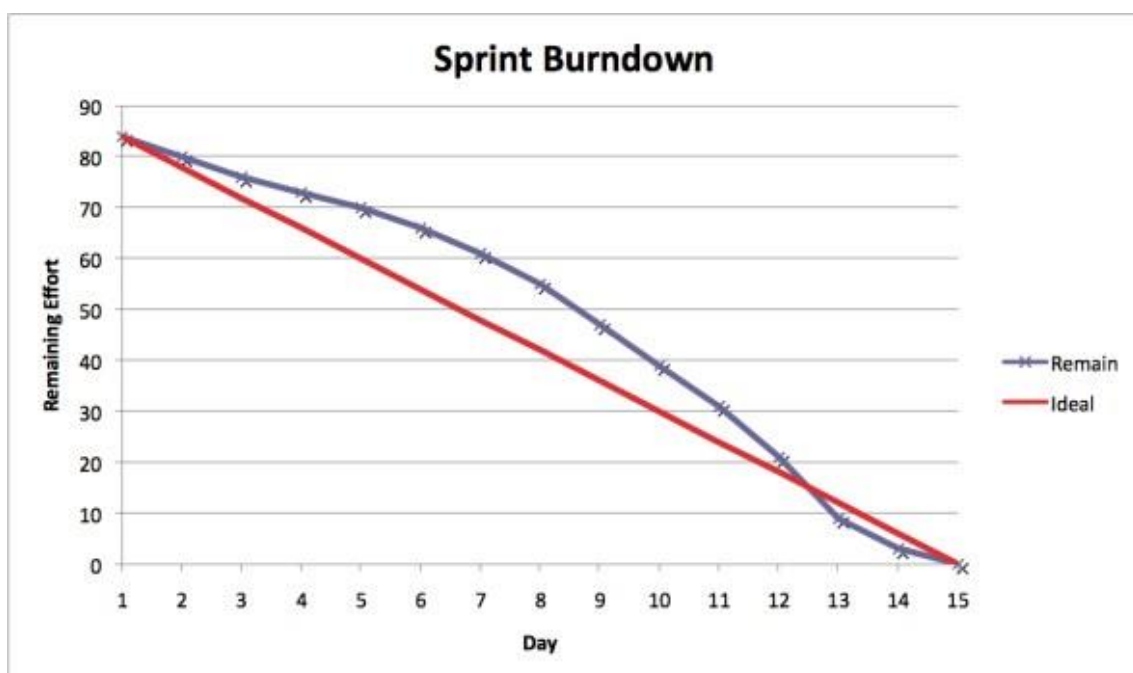


Ilustración 4.1. Gráfico burn down

A partir, del gráfico podemos sacar muchas conclusiones. Vamos a estudiar la imagen de ejemplo.


Para el día 7, lo ideal sería que quedasen algo menos de 50 story points por terminar. Sin embargo, aún quedan 60 story points. Vemos como durante el día 12 la cosa mejora e incluso se incrementa la velocidad de trabajo al final.

Por el bien del equipo es aconsejable que la línea de esfuerzo real se aproxime lo máximo posible a la línea ideal. Ya que terminar todo el trabajo al final puede generar situaciones de estrés y conflictos internos, el que equipo llega cansando a sprints posteriores y éstos pueden verse afectados y además, si los programadores no entregan las historias de usuario de manera parcial los testers pueden verse sobrepasados al final del sprint.

Como ya hemos comentado, un gráfico burn-down es una representación gráfica del trabajo que queda por hacer frente al tiempo.

En la representación común el trabajo está en el eje vertical, y el tiempo a lo largo de la horizontal.

Imagina que eres Scrum Master y tienes que crear el gráfico de manera manual: **¿Cómo creas un gráfico de Burndown?**



Ejemplo

Vamos a verlo con un ejemplo:

- | **Imaginemos un sprint con una duración: 5 días**
- | **Un Sprint Backlog con 8 tareas**
- | **Y una velocidad para el equipo de 80 horas disponibles este sprint.**

Paso 1 –Trazar el esfuerzo estimado

Imagina tu línea de base ideal para usar las horas disponibles durante el sprint.

Entonces, lo más simple para esto son las horas disponibles divididas por el número de días. En este ejemplo, 80 horas durante 5 días equivalen a 16 horas al día.

Para crear el gráfico, los datos deben capturarse como un total diario que comienza con 80 horas, al final de día 1 quedan 64 horas(80 - 16), al final del día 2 quedan 48 horas, etc.

Paso 2 - Seguimiento del proceso diario

El progreso diario se captura en la tabla con cada tarea.

Es importante recordar que el valor capturado para cada día es el esfuerzo estimado para completar la tarea, no el esfuerzo real.

Paso 3 - Calcular el esfuerzo real

El esfuerzo restante total debe ser capturado al final de cada día.

Éste es el total (la suma) de todo el tiempo estimado restante al final de cada día.

Paso 4 - Obtener el conjunto de datos final

Cuando los datos están disponibles, se puede crear el gráfico burn-down.

Esto es relativamente simple usando la opción de gráfico de líneas disponible en Excel.

La tabla de resumen contiene el total diario del esfuerzo de referencia y el esfuerzo estimado.

Y el encabezado el período de tiempo (Día 0, Día 1, etc).

Paso 5: Dibujar el gráfico Burndown

Usando el conjunto de datos de la tabla y una herramienta como Excel o similar puedes dibujar el gráfico Burndown.

Como ves es muy sencillo crear un gráfico siguiendo estos pasos, siempre que sepas qué representan y cómo utilizarlos.

Por suerte para ti, existen herramientas de gestión de proyectos ágiles como vimos en la primera lección, que realizan este trabajo de manera automática para que no tengas que hacer el trabajo de manera manual cada sprint.

6. HISTORIAS DE USUARIO

En el ámbito del desarrollo de software, los equipos ágiles gestionan los requisitos del cliente, escribiéndolos en pilas de product mediante historias de usuario, mientras que los equipos tradicionales usan documentos SRS (Software Requirements Specification).

En la jerarquía de los requisitos ágiles se encuentran los epics, los temas y las historias de usuario. Tienen en común que todos se centran en describir qué se va a construir y se pueden representar gráficamente en función de la granularidad de la pila de product.

La granularidad es el nivel de detalle de cada elemento en la pila de product. Las epics tienen una granularidad alta o gruesa; las tareas, una granularidad baja o fina.

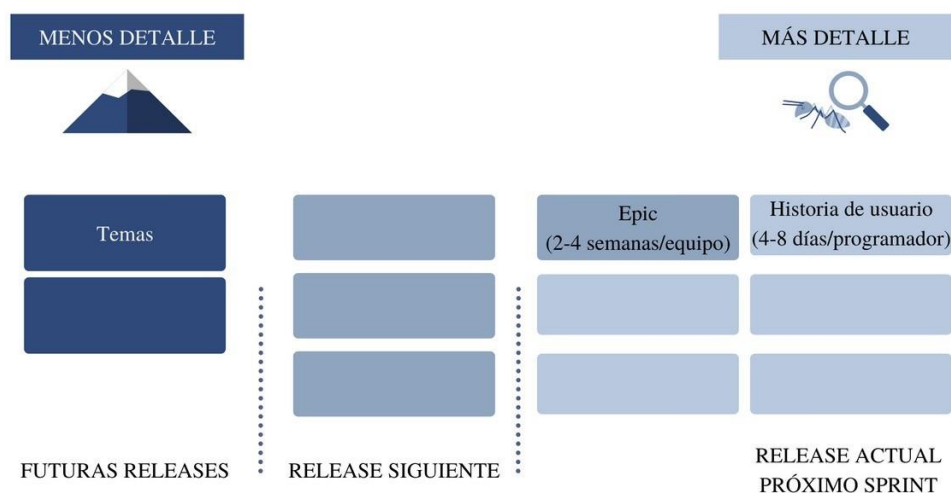


Ilustración 5.1. Granularidad de la pila de product

A mayor prioridad, menor granularidad.

La ingeniería de requisitos dispone de dos medios de comunicación para recoger y transmitir requisitos, cada uno con sus características:

Comunicaciones escritas.

1. Registran la información de forma permanente.
2. Son más fáciles de compartir con grupos y personal remoto.
3. Se pueden pensar y revisar bien y de forma completa.
4. Son más susceptibles de malinterpretarse.

Comunicaciones verbales.

1. Permiten recibir feedback de inmediato.
2. Son dinámicas; la conversación se adapta para maximizar su eficiencia.
3. Se adaptan con facilidad a nuevos desarrollos.
4. Generan ideas nuevas.
5. Permiten alcanzar comprensión y claridad comunes con menos esfuerzo.

Las historias de usuario se usan, en el contexto de la ingeniería de requisitos ágil, como una herramienta de comunicación que combina las fortalezas de ambos medios: escrito y verbal. Describen, en una o dos frases, una funcionalidad de software desde el punto de vista del usuario, con el lenguaje que éste emplearía. El foco está puesto en qué necesidades o problemas soluciona lo que se va a construir.

Las historias de usuario son una herramienta que agiliza la administración de requisitos, reduciendo la cantidad de documentos formales y tiempo necesarios. Forman parte de la fórmula de captura de funcionalidades definida en 2001 por Ron Jeffries de las tres Cs:

- | **Card:** cada historia de usuario se reduce hasta hacerla fácil de memorizar y de sintetizar en una tarjeta o post-it. La tarjeta sirve como recordatorio y promesa de una conversación posterior.
- | **Conversation:** el equipo de desarrollo y el propietario del producto añaden criterios de aceptación a cada historia poco antes de su implementación. Los cambios son bienvenidos en agilidad, por lo que no tiene sentido profundizar en estos detalles antes. La situación puede variar mucho desde el momento en el que se sintetiza la funcionalidad en la tarjeta hasta que se implementa.
- | **Confirmation:** el propietario del producto o usuario de negocio confirma que el equipo de desarrollo ha entendido y recogido correctamente sus requisitos revisando los criterios de aceptación. A veces se pueden presentar transformados en escenarios de pruebas.

Ventajas de las historias de usuario:

1. Proporcionan la documentación necesaria fomentando a la vez el debate.
2. Fomentan la colaboración entre todos interesados y el equipo ágil.
3. Se escriben en el lenguaje del usuario, manteniendo así una relación cercana con el cliente.
4. Involucran y captan al cliente para el proceso y para el producto.
5. Por su naturaleza son independientes.
6. Facilitan la planificación e implementación.
7. Son ideales para proyectos con requisitos volátiles o no muy claros.
8. Fomentan aplazar los detalles no imprescindibles.
9. Son pequeñas y por tanto fáciles trabajar.
10. Permiten dividir los proyectos en pequeñas entregas.
11. Permiten estimar fácilmente su esfuerzo de desarrollo.
12. Funcionan para el desarrollo iterativo, ya que al ser pequeñas representan requisitos del modelo de negocio que pueden implementarse en poco tiempo (días o semanas).
13. Necesitan poco mantenimiento.

6.1. Información en las historias de usuario

Para decidir qué información incluir en una historia de usuario es preferible no adoptar formatos rígidos. Los resultados de scrum y agilidad no dependen de las formas, sino de la institucionalización de sus principios y la implementación adecuada a las características de la empresa y del proyecto.

Por tanto, aparte de tres campos que se consideran necesarios, se puede incluir cualquier campo que proporcione información útil para el proyecto.

El objetivo de las historias de usuario es construir un entendimiento compartido.

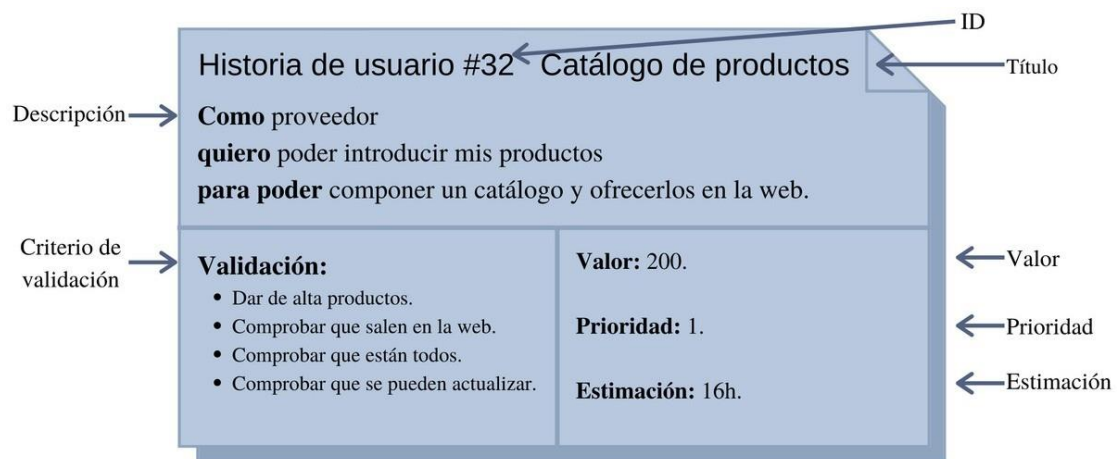


Ilustración 6.2. Ejemplo de tarjeta de historia de usuario

Los campos esenciales de las historias de usuario son:

Descripción: síntesis de la historia de usuario. El estilo puede ser libre pero debe responder a tres preguntas: ¿quién se beneficia? ¿qué se quiere? y ¿cuál es el beneficio?

Mike Cohn (uno de los padres de Scrum) recomienda seguir el siguiente patrón para garantizar que la funcionalidad esté descrita a un alto nivel y de manera breve.

Como [rol del usuario], **quiero** [objetivo], **para poder** [beneficio].

Estimación: aproximación del esfuerzo necesario (en tiempo ideal) para implementar la historia de usuario. Suele estimarse usando unidades de desarrollo como puntos de historia (veremos este concepto más adelante).

Prioridad: se indica siguiendo un sistema que permita establecer el orden de implementación de las historias.

Dependiendo del tipo de proyecto, el funcionamiento del equipo y la organización, pueden ser aconsejables otros campos como:

- | ID: identificador único de la historia de usuario, funcionalidad o trabajo.
- | Título: título descriptivo de la historia de usuario.
- | Valor de negocio: valor (normalmente numérico) que aporta la historia de usuario al cliente o usuario. El objetivo del equipo es maximizar el valor y la satisfacción percibida por el cliente en cada iteración. Este campo servirá junto con la estimación para decidir la prioridad de implementación.
- | Criterio de aceptación: pruebas de aceptación consensuadas con el cliente o usuario. A veces se transforman en pruebas que el código debe superar para dar como finalizada la implementación.
- | Requerimiento no funcional: cualidades generales y restricciones, como la usabilidad o la seguridad, que afectan a aplicaciones y sistemas enteros, y por tanto a las historias de usuario individuales.
- | Definición de hecho (en inglés DoD, Definition of Done): incluye las actividades o criterios necesarios para dar por terminada una historia de usuario (desarrollada, probada, documentada...), según lo convenido por el propietario de producto y el equipo.
- | Dependencias: una historia de usuario no debería depender de otra, pero a veces es necesario mantener la relación. Este campo contendría los identificadores de otras historias de las que depende.
- | Persona asignada: cuando queramos sugerir la persona que pueda implementar la historia de usuario. Recordar que en scrum el equipo se autogestiona y es quien distribuye y asigna las tareas.
- | Sprint: puede ser útil para organización del propietario del producto incluir el número de sprint en el que se prevé construir la historia.
- | Riesgo: riesgo técnico o funcional asociado a la implementación de la historia de usuario.
- | Módulo: módulo del sistema o producto al que pertenece.
- | Observaciones: para enriquecer o aclarar la información, o cualquier otro uso necesario.

7. PUNTOS CLAVE

- | Artefactos de Scrum.
- | Historias de usuario como artefacto fundamental de Scrum.

